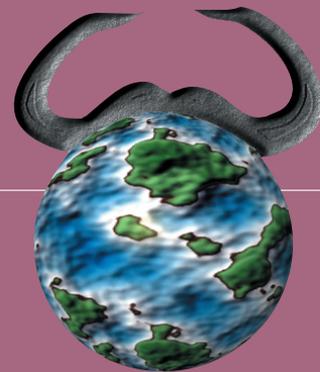


# Brave GNU World



Diese Kolumne berichtet aus der Perspektive von GNU-Projekt und FSF über Projekte und aktuelle Geschehnisse aus dem Umfeld freier Software und versucht Einblicke in die zugrunde liegende Philosophie zu vermitteln. In dieser Ausgabe geht es vor allem um Programme zur Kommunikation mit Mobiltelefonen. Georg C. F. Greve



Abbildung 1: Gnocky ist Gnokii's neue grafische Schnittstelle zum Mobiltelefon, mit der sich alle wichtigen Daten abfragen lassen.

**Willkommen** zu einer weiteren Ausgabe der Brave GNU World, in der es diesmal unter anderem um freie Software im mobilen Einsatz mit UMTS, GPRS, SMS & Co. geht. Unbemerkt von vielen Nutzern entwickelte sich in den letzten Jahren hier eine große Vielfalt.

## Gnokii

Gnokii [5] gehört sicherlich zu den bekanntesten freien Projekten für die Kommunikation mit Mobiltelefonen. Es entstand im Februar 1999 aus dem Zusammenschluss zweier Projekte, die die Funktionalität der Nokia Cellular Data Suite (CDS) auf anderen Plattformen, insbesondere auf Linux zur Verfügung stellen wollen.

Das eine Projekt bestand aus Interessierten um Francois Dessart und Hugh Blemings, die im Oktober 1998 mit der Entwicklung begannen. Staffan Ulfberg initiierte das andere und hatte bis dahin

noch keinen Code produziert. Mit gebündelten Kräften machte die Arbeit jedoch große Fortschritte, bis das Projekt in der zweiten Hälfte des Jahres 2000 ins Straucheln kam: Pavel Janik und Hugh Blemings, zwei Hauptentwickler, konnten nur noch wenig Zeit einbringen. Dank der aktiven Gemeinschaft war es jedoch möglich, dass andere Freiwillige die Betreuung des Projekts übernahmen, und so ist Gnokii mittlerweile weit entwickelt.

Das Projekt besteht im Wesentlichen aus drei Anwendungen: einem Gnokii-Daemon, der mit Supernutzer-Rechten im Hintergrund läuft, einer Kommandozeilen-Anwendung sowie einer Anwendung mit grafischer Benutzeroberfläche. Eine Hilfsapplikation vervollständigt die Programmsuite. Sie sucht bei Bedarf automatisch das Mobiltelefon als Modem für die Einwahl.

Geschrieben in C und als freie Software unter der GNU General Public License (GPL) veröffentlicht, unterstützt Gnokii mittlerweile eine ziemlich große Zahl an Telefonmodellen, speziell die Nokia-Serien 3810, 6110, 7110 und 6510. Daneben sind einigen Telefone von Bosch, Siemens, Motorola und Sony-Ericsson ebenfalls kompatibel, wenn sie mittels AT-Kommandos steuerbar sind.

Die Programmsuite erlaubt die Einwahl über das Mobiltelefon ins Internet ebenso wie das Sichern des Telefonbuchs in einer Textdatei auf der Festplatte oder das Retten des Telefonbuchs in umgekehrter Richtung. Zusätzlich las-

sen sich Kurznachrichten bequem über eine Frontend verfassen oder lesen. Wem die üblichen Logos nicht individuell genug sind, dem bietet Gnokii die Möglichkeit, eigene Icons mit Hilfe eines kleinen Editors zu erstellen und ins Telefon hochzuladen.

Da Gnokii in den meisten Distributionen bereits enthalten ist, bereitet die Installation keine größeren Probleme. Für die weniger Furchtsamen bietet sich zudem die Chance, zwei neuere Entwicklungen auszuprobieren: Es gibt auf der Gnokii-Webseite [5] die neue grafische Benutzeroberfläche Gnocky sowie das Gnokii-Applet »gnapplet«, beide sind noch nicht umfangreich getestet und erfordern daher Handarbeit.

## MyGnokii

Mit MyGnokii [6] teilte sich das Gnokii-Projekt in der zweiten Hälfte des Jahres 2000. Damals ging das ursprüngliche Projekt gerade durch schwierige Zeiten. Hauptautor von MyGnokii wurde Marcin Wiacek, der es gemeinsam mit Gabriele Zappi, Ralf Thelen und anderen weiterentwickelte.

Der Fork eines Projekts wird oft als Nachteil freier Software beschrieben. Dabei stellt er in Wahrheit eine große Stärke dar. Historisches Beispiel ist die Gabelung von GCC und EGCS, die mit einer Wiedervereinigung und erheblich verbesserten C++-Fähigkeiten des GCC endete. Wenn sich in einem Projekt Gruppen bilden, die einerseits ausreichend motiviert und andererseits ausreichend uneins sind, kommt es vor, dass diese Gruppen das Projekt unterschiedlich weiterentwickeln und nicht selten in Konkurrenz zueinander stehen.

Das gibt nicht nur beiden Ansätzen eine echte Chance, sich zu beweisen und herauszufinden, welcher der Bessere ist, sondern führt oft zur Wiedervereinigung oder zumindest zu einem Quertransfer von Konzepten und Ideen.

## Problemfall Lizenz

In diesem Fall hat Marcin Wiacek beklagt, dass Gnokii Code aus MyGnokii übernommen hat. Denn Gnokii steht unter der GPL-Version 2 oder einer späteren Version nach Wahl des Nutzers, MyGnokii aber explizit unter der GPL-Version 2, was das Gnokii-Team nach seiner Aussage nicht respektiert hat.

Auch aus dem Minicom-Projekt, einer Terminalsoftware, soll mit demselben Problem Sourcecode in Gnokii integriert worden sein. Hier scheinen zusätzlich die Copyright-Vermerke beim Schritt von Minicom zu Gnokii verloren gegangen zu sein. Einige Leute sprechen daher von einer Verletzung der GPL. Das Debian-Projekt beschloss aus diesem Grund, das Programm Gnokii aus der Distribution zu nehmen, bis sich die rechtliche Situation zwischen den Projekten ausreichend geklärt hat.

Dabei handelt es sich nach dem aktuellen Stand um eine Copyright- und keine GPL-Verletzung: Das Entfernen des Copyright-Vermerks oder das Hinzufügen der Klausel „... or any later version“ sind keine Probleme der Lizenz. Am Ende ist zumindest eine Nutzung der Software unter der GPL-Version 2 möglich.

Debian hätte also theoretisch beschließen können, das Programm ausdrücklich unter der GPL 2 zu veröffentlichen, wenn die Frage des entfernten Copyright-Hinweises nicht gewesen wäre. Es spricht ausdrücklich für das Debian-Projekt, dass es so große Mühe darauf verwendet, die Distribution ohne Zweifel rechtlich sauber zu halten.

Trotz solcher Schwierigkeiten profitiert der Anwender in der Regel von einem Fork, denn der hier beobachtbare Transfer von Sourcecode ist in vieler Hinsicht ein sehr vernünftiges Phänomen, bei dem eine gewisse Konsolidierung des MyGnokii-Zweiges in den Gnokii-Zweig beabsichtigt zu sein scheint.

## Aus MyGnokii wird Gammu

Im April 2003 beschlossen die Entwickler, das Projekt MyGnokii einzustellen. Da das interne Design zunehmend zum Hindernis für die weitere Arbeit geriet, nahmen sie eine vollständige Neuentwicklung in Angriff, etwas später erhielt das Projekt den Namen Gammu [7]. Wie seine Vorläufer MyGnokii und Gnokii ist Gammu im Wesentlichen in C geschrieben und wird von Marcin Wiacek als freie Software unter der GPL herausgegeben. Gemeinsam mit Michal Cihar, Waldek, Joergen Thomsen und vielen anderen entwickelte er ein sehr funktionales und flexibles Programm.

So unterstützt die Software Telefone mehrerer Hersteller sowie Verbindungen über verschiedene Kabel, Infrarotschnittstellen oder Bluetooth-Stacks. Zudem unterstützt Gammu VCards ebenso wie den Versand und den Empfang von SMS und MMS. Telefonbücher inklusive assoziierter Informationen wie Typ der Telefonnummer oder Rufnummerngruppe lassen sich auslesen und schreiben. Mit Hilfe des OBEX-Protokolls – eines HTTP-ähn-

lichen Binärprotokolls – ist bei vielen Telefonen der Zugriff auf Bilder, Klingeltöne, Stimmaufzeichnungen, Java-Programme, E-Mails oder den Anrufbeantworter lesend und zum Teil auch schreibend möglich.

Ein Netzwerk-Monitoring-Modul beobachtet die Qualität des Netzes, mit der Diagnosefunktion spürt der Benutzer Fehler auf. Die Sicherung und Wiederherstellung übernimmt ein Backup-Modul. Die Liste der Features ist erstaunlich lang und die Entwicklung noch immer aktiv. Allerdings gibt es inzwischen umso mehr Schwierigkeiten, je jünger die Telefone sind.

Verantwortlich für dieses Problem ist vor allem die mangelnde Unterstützung durch die Hersteller. Das Gnokii-Projekt berichtet von ähnlichen Schwierigkeiten. In der Folge müssen die Autoren sich ihre Hardware und viele Informationen selber beschaffen, was bei dem knappen Zeit- und Finanzbudget teilweise zu erheblichem Druck führt.

Dieser Druck war maßgeblich auch dafür verantwortlich, dass Marcin Wiacek die Lizenz von General Public License (GPL) auf Lesser General Public License (LGPL) ändern wollte, um die proprietäre Nutzung zu ermöglichen. Davon versprach er sich eine bessere Finanzierung. Das Unterfangen scheiterte allerdings an der fehlenden Zustimmung der anderen Entwickler.

Aus dieser Kontroverse heraus entschloss sich Marcin, parallel an einer Reimplementation von Gammu in C++ zu arbeiten. Auf eine Lizenz für dieses Projekt hat er sich gegenwärtig noch nicht festgelegt, plant allerdings eine proprietäre Lizenz. Genauer: Gammu + ist zurzeit frei für die nicht kommerzielle Nutzung, nach der Definition für freie Software („Unbegrenzte Nutzung für jeden Zweck“) jedoch klar unfrei. Das ist ein bedauerlicher Schritt, nicht zuletzt weil sich bisher noch kein Interessent einer proprietären Nutzung gefunden zu haben scheint.

Letztlich ist es sehr verständlich, dass Marcin nach einem Weg sucht, um die weitere Entwicklung und seine Zeit an diesen Projekten zu finanzieren, und es ist sehr schade, dass so oft die Nutznießer der Arbeit nicht sicherstellen, dass diese weiterhin möglich ist. Erfolgreiche

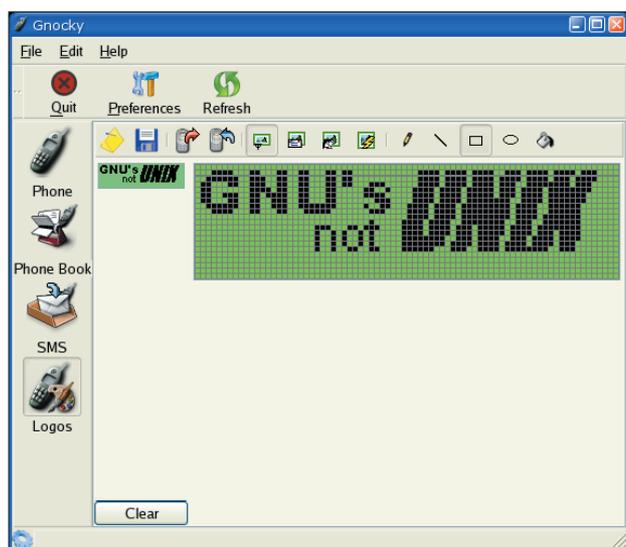


Abbildung 2: Mit einigen Programmen lassen sich die Logos im Handy verändern.

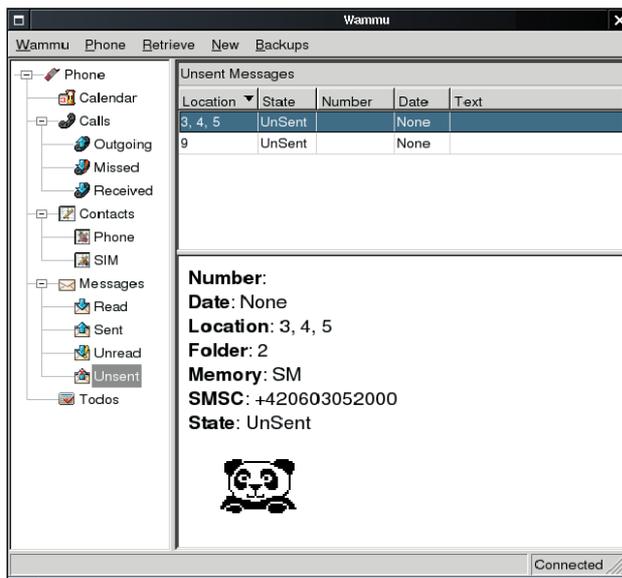


Abbildung 3: Wammu ermöglicht komfortablen Zugriff auf Handy-Daten.

Projekte holen aus den Telefonen einen Mehrwert für den Anwender heraus und sorgen für mehr Umsatz bei den Netzbetreibern. Hier ist mit wenig Aufwand viel zu erreichen – dies nicht zu unterstützen ist kurzsichtig und dumm.

Doch trotz aller Einsicht ist der Schritt in Richtung proprietärer Software aus vielen Gründen schlecht. Zum einen ist der Erfolg keinesfalls so garantiert, wie es scheinen mag. Ein proprietäres Programm ist nicht attraktiver – das Gegenteil ist häufig der Fall. Die Nutzer wenden sich im Laufe der Zeit wieder freien Projekten zu, aber das eigene Projekt stagniert inzwischen.

## Dual-Lizenzierung

Wer Teile aus den freien Projekten in eigenen unfreien Projekt nachprogrammiert, setzt sich schnell dem Verdacht einer Copyright-Verletzung aus. Dies ist umgekehrt übrigens der Grund, warum im GNU-Projekt bewusst niemand ein Programm nachprogrammiert, für dessen proprietäres Gegenstück er Einblick in den Sourcecode hatte. Im Gammu-Projekt kommt es derzeit in Fragen des Copyrights und der GPL zwischen einigen Beteiligten zusätzlich zu Verstimmungen.

Grundsätzlich darf der Copyright-Inhaber beziehungsweise der Urheber eines Programms es unter beliebig vielen Lizenzen herausgeben. Am gebräuchlichsten ist dabei die Kombination einer

müssen, sofern ihre Änderungen in die Hauptversionen Einzug halten sollen. Andernfalls wären die Unternehmen aus rechtlichen Gründen dazu gezwungen, den Code von Dritten aus den unfreien Versionen zu entfernen.

Unternehmen verkaufen also die Arbeit Dritter proprietär, ohne sie am Gewinn zu beteiligen. Zwar bemühen sie sich wie beispielsweise im Falle von MySQL darum, unfreie Änderungen in den freien Zweig zu übertragen. Das ist jedoch zumindest eine Modifikation des üblichen Modells der Solidarität. Die übertragenen Rechte bleiben übertragen, das bedeutet, dass der externe Entwickler großes Vertrauen in die moralische Integrität eines Unternehmens haben muss, wenn er nicht möchte, dass seine Arbeit verloren geht oder gegen freie Software eingesetzt wird.

Dagegen wäre Gammu+ ein Projekt, neben dem bereits jetzt mit Gnokii und Gammu zwei starke freie Konkurrenten bestehen, die sich eventuell sogar vereinigen könnten. Mit Hilfe von geschickt konstruierten Wrappern könnten Hersteller unfreier Software diese nutzen – und damit Zahlungen an Gammu+ vermeiden. Statt eines proprietär entwickelten bekämen sie ein aktiv vorangetriebenes freies Projekt.

Die Lösung aus diesem Dilemma könnte etwas anders aussehen: Mehr Bekanntheit, eine stärkere Zusammenarbeit mit der Gemeinschaft und die Möglichkeit von Online-Spenden würden helfen, das

freien Software-Lizenz und einer proprietären – das wird dann Dual- oder auch Doppel-Lizenzierung genannt.

Oft zitierte Beispiele sind Open Office und MySQL: Beide Programme liegen in einer solchen Dual-Lizenzierung vor. Das bedeutet für extern Mitwirkende aber, dass sie ihre Rechte an Sun oder die Firma

MySQL übertragen

Projekt voranzubringen. Auch attraktivere und professionellere Webseiten können mehr Leute anlocken und außerdem dabei helfen, den Unternehmen, die die Entwicklung unterstützen sollen, Seriosität zu demonstrieren.

Wammu [8] schließlich ist ein grafisches Frontend für Gammu, das Michal Cihar geschrieben hat und ebenfalls als freie Software unter der GPL herausgibt. Es bietet auf Basis von WX Python eine ansprechende grafische Oberfläche für die zentralen Features von Gammu: Arbeit an den Adress- und Todo-Listen sowie am Kalender, das Senden und Empfangen von SMS inklusive der Wiedergabe von Bild und Ton sowie eine Backup-Funktion.

Dank der vorbereiteten Pakete für Suse und Debian dürfte es den meisten Nutzern möglich sein, Wammu einigermaßen zügig zu installieren und so entweder mit Gnocky oder Wammu bequem in die Welt der mobilen Kommunikation einzutreten. (agr)

## Infos

- [1] Ideen, Anregungen, Kommentare an die Brave GNU World: [\[column@brave-gnu-world.org\]](mailto:column@brave-gnu-world.org)
- [2] Homepage des GNU-Projekts: [\[http://www.gnu.org/\]](http://www.gnu.org/)
- [3] Homepage von Georg's Brave GNU World: [\[http://brave-gnu-world.org\]](http://brave-gnu-world.org)
- [4] „We run GNU“ Initiative: [\[http://www.gnu.org/brave-gnu-world/rungnu/rungnu.de.html\]](http://www.gnu.org/brave-gnu-world/rungnu/rungnu.de.html)
- [5] Gnokii-Homepage: [\[http://www.gnokii.org\]](http://www.gnokii.org)
- [6] MyGnokii-Homepage: [\[http://www.mwiacek.com/gsm/soft/mygnokii.html\]](http://www.mwiacek.com/gsm/soft/mygnokii.html)
- [7] Gammu-Homepage: [\[http://www.mwiacek.com/gsm/soft/gammu.html\]](http://www.mwiacek.com/gsm/soft/gammu.html)
- [8] Wammu-Homepage: [\[http://www.cihar.com/gammu/wammu/\]](http://www.cihar.com/gammu/wammu/)

## Der Autor

Dipl.-Phys. Georg C. F. Greve beschäftigt sich seit etlichen Jahren mit freier Software und kam früh zu GNU/Linux. Nach Mitarbeit im GNU-Projekt und seiner Aktivität als dessen europä-



ischer Sprecher hat er die Free Software Foundation Europe initiiert, deren Präsident er ist. Mehr Informationen finden sich unter: [\[http://www.gnuhh.org\]](http://www.gnuhh.org)